Comparison of Routing Software in Linux

Master’s Thesis Presentation
10.10 2006

Tuukka Taipale
Nokia NET / Core Networks

Supervisor: Professor Raimo Kantola

Presentation Contents

1. Overview
2. Objective
3. Methods
4. Results
5. Conclusion
1. Overview

The purpose of routing:
A Router needs to forward the Internet Protocol packets towards the destination

Routing software:
An application that performs the routing tasks

Routing tasks:
1. Keep the routing information database up to date
2. Forward the packets to the right direction
2. Objective

- We have a good routing software in FreeBSD
- We need a routing software in Linux
- Measure how well the FreeBSD software works in Linux

3. Methods

- Migrate the routing software from FreeBSD to Linux
- Compare the migrated routing software to a few other routing software packages in Linux
- Comparison consists of
  - Performance and
  - Software complexity measurements
Compared routing software packages:

- Nokia ipsrd
- IpInfusion ZebOS
- NextHop GateD
- Quagga

Performance comparison contents:

- Insertion time of a big amount routes
- Memory usage
- OSPF convergence time
Software complexity measurement contents:

- Lines-of-code metrics
- Cyclomatic complexity measurements
- Information volume
- Maintainability Index

4. Results

- Ipsrd works well also in Linux
- ZebOS and ipsrd are on about the same level but better than the others
- GateD and Quagga are also equally good
Performance results:

- No big differences in route insertion time
- Ipsrd and GateD use significantly less memory than ZebOS and Quagga
- OSPF convergence times are much smaller with ZebOS and Quagga than with ipsrd and GateD

Complexity measurement results:

- Lines-of-code metrics:
  No big differences between the routing solutions. Ipsrd get slightly the best points, GateD is the second. ZebOS and Quagga are equally good.

- Cyclomatic complexity measurements:
  Quagga and GateD are the best. Ipsrd and ZebOS are worse and on about the same level.
Complexity measurement results:

- Information volume:
  No big differences between the routing solutions. ZebOS is slightly the best. Ipsrd and Quagga share the second place.

- Maintainability Index:
  Ipsrd gets the first place in this measurement. ZebOS and GateD share the second place. Quagga is the most difficult to maintain.

5. Conclusion

- The initial objectives were fulfilled in the thesis

- Ipsrd placed well in the comparison results even if it was only a pilot version

- Future work is needed to optimize ipsrd for Linux
Thank You!

Something to clarify?

Performance Results

<table>
<thead>
<tr>
<th>Time to insert 100000 OSPF routes</th>
<th>OSPF Convergence time</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Nokia Ipsrd</strong></td>
<td><strong>IpInfusion ZebOS</strong></td>
</tr>
<tr>
<td><strong>NextHop Gated</strong></td>
<td><strong>Quagga</strong></td>
</tr>
<tr>
<td><strong>time (s)</strong></td>
<td><strong>time (s)</strong></td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>100</td>
<td>8.10</td>
</tr>
<tr>
<td>1000</td>
<td>8.57</td>
</tr>
<tr>
<td>10000</td>
<td>10.94</td>
</tr>
<tr>
<td><strong>Routes</strong></td>
<td><strong>time (s)</strong></td>
</tr>
<tr>
<td>100</td>
<td>9.67</td>
</tr>
<tr>
<td>1000</td>
<td>12.57</td>
</tr>
<tr>
<td>10000</td>
<td>34.54</td>
</tr>
</tbody>
</table>
Complexity Results 1/2

Tuukka Taipale
10.10.2006
17

Complexity Results 2/2

Tuukka Taipale
10.10.2006
18
## Comparison Summary

### Weighted comparison results

<table>
<thead>
<tr>
<th></th>
<th>Nokia Ipsrd</th>
<th>IpInfusion ZebOS</th>
<th>NextHop GateO</th>
<th>Quagga</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Performance</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inserting 100000 OSPF routes</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Memory Usage Average</td>
<td>2.5</td>
<td>1</td>
<td>2.5</td>
<td>0</td>
</tr>
<tr>
<td>OSPF Convergence</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td><strong>Sum</strong></td>
<td>3.3</td>
<td>6</td>
<td>3.5</td>
<td>5</td>
</tr>
<tr>
<td><strong>Complexity Measurements</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maintainability Index</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Double MI</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td><strong>Sum</strong></td>
<td>9.5</td>
<td>10</td>
<td>5.5</td>
<td>5</td>
</tr>
</tbody>
</table>